

Министерство образования и науки Российской Федерации  
ФГБОУ ВПО «Уральский государственный педагогический университет»  
Институт математики, физики, информатики и технологий  
Кафедра физики и математического моделирования

## **Разработка и создание программного обеспечения для IT службы организации**

Выпускная квалификационная работа

Квалификационная работа  
допущена к защите  
Зав кафедрой ФиММ  
д. ф.-м. н., профессор  
Сидоров Валерий Евгеньевич

\_\_\_\_\_  
Дата

\_\_\_\_\_  
Подпись

Исполнитель:  
Задорин Максим Алексеевич,  
обучающийся группы БЭ 51Z

\_\_\_\_\_  
Подпись

Научный руководитель  
Минина Елена Евгеньевна  
кандидат педагогических наук  
доцент кафедры ФиММ

\_\_\_\_\_  
Подпись

Екатеринбург 2017

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	3
1 АНАЛИЗ, РАЗРАБОТКА ЭСКИЗНОГО И ТЕХНИЧЕСКОГО ПРОЕКТА ПРОГРАММЫ .....	6
1.1 Анализ, информационной структуры предприятия и определение основной группы проблем .....	6
1.2 Назначение и область применения .....	9
1.3 Технические характеристики. Постановка задачи. Спецификация технических и программных средств .....	10
1.4 Набор инструментов разработчика Embarcadero RAD Studio XE6 Delphi: назначение и общее описание среды. Структура среды программирования .....	13
1.5 Базы данных и системы управления базами данных .....	17
1.6 Разработка HTTP Web-страницы на основе HTML и PHP5, описание языка web-программирования. Описание Sendmail.....	23
1.7 Шифрование и дешифрование конфиденциальных данных посредством ключа ..	26
2 РАЗРАБОТКА КОМПОНЕНТОВ СИСТЕМЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ ИТ СЛУЖБЫ ПРЕДПРИЯТИЯ.....	29
2.1 Разработка базы данных .....	29
2.1.1. Web-форма отправки сообщений-заявок .....	30
2.1.2 Временные и постоянные переменные .....	32
2.1.3 Внешние и внутренние компоненты, Файл конфигурации, шифрование паролей пользователя.....	32
2.1.4 Интерфейс программы RemoteNetworkTracer .....	34
2.2 Спецификация программы RemoteNetworkTracer .....	36
2.3 Описание программы RemoteNetworkTracer .....	38
2.4 Тестирование, Внедрение и функционал.....	40
2.6 Выполнение программы. ....	41
2.6.1 Функциональные кнопки блока «удаленное управление» и блока Putty .....	42
2.6.2 Блок управления БД, реквизитами авторизации, настройки электронной почты ..	43
ЗАКЛЮЧЕНИЕ .....	48
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	50
ПРИЛОЖЕНИЕ 1 .....	53
ПРИЛОЖЕНИЕ 2 .....	54
ПРИЛОЖЕНИЕ 3 .....	55

## **ВВЕДЕНИЕ**

В наше время в большинстве стран мира растет потребность в вычислительных устройствах. Вычислительная электроника является универсальным устройством для обработки огромного количества информации. Она может выполнять любые действия по получению, хранению, обработке и передачи информации. Для выполнения поставленных задач, нужно составить на понятном для исполнителя языке логичную и подробную последовательность инструкций – программное приложение, как надо обрабатывать информацию.

Разрабатывая новое программное обеспечение, можно решать множество практических и научных задач, решать сложные математические или логические уравнения, последовательности, рассчитывать прибыль и расходы предприятия или просто разработать приложение для развлечения или решения простых повседневных задач пользователей. Помимо этого, тенденция снижения стоимости компьютерной техники при одновременном росте ее производительности привела к тому, что компьютеры и компьютеризированные устройства стали обычным домашним устройством, которое присутствует практически у каждого жителя нашей страны, например, холодильник или телевизор, телефон, планшет, и многое другое, что расширяет сферу применения вычислительной техники. Соответственно, с ростом спроса и потребности людей в цифровой электронике, в автоматизации процессов, систематизации и обработке информации требуется все более разнообразное программное обеспечение для решения новых узкоспециализированных задач.

В настоящее время весь комплекс программного обеспечения делится на системные и пользовательские программы. Системное программное обеспечение выполняет функции управления всеми устройствами цифрового оборудования, создавая платформу для пользовательского программного обеспечения, а также подключенных к нему внешних устройств.

Пользовательское программное обеспечение служит для выполнения, каких – либо конкретных задач во всех сферах человеческой деятельности.

Написание выпускной квалификационной работы является важнейшей частью и неотъемлемой ступенью для формирования квалифицированного специалиста, будущего выпускника учебного заведения. Разработка и понимание принципов создания программных продуктов, даёт реальную возможность обобщить и систематизировать свои знания в области фундаментальных и прикладных наук и направить их на самостоятельное решение комплекса управленческих задач.

Актуальность написания работы обусловлена необходимостью обобщения, систематизации, закрепления и углубления теоретических знаний, полученных в течение обучения в ИМФИиТ УРГПУ г. Екатеринбург, и применения их в практической деятельности применительно к специальности и профилю будущей профессии.

Объектом исследования является МБОУ СОШ №20 г. Нижний Тагил. Предметом исследования является анализ текущей программно-информационной структуры организации, и программных средств автоматизирующих основные функции ИТ администратора, обмен сообщениями между пользователем и обслуживающим специалистом, оптимизация времени работы сотрудников ИТ службы при решении повседневных задач.

Целью выпускной квалификационной работы является разработка пользовательской программы для специалистов информационной поддержки в ИТ службе образовательного учреждения, средняя образовательная школа №20.

Для осуществления цели необходимо выполнить следующие задачи:

1. Провести анализа информационной структуры учреждения.
2. Определить основной группы проблем.
3. Определить список требований к программному обеспечению.
4. Провести анализ рынка продуктов. Поиск программ аналогов.

5. Определить набор инструментов разработчика.
6. Разработать, протестировать, внедрить программное обеспечение для ИТ службы учреждения.

# **1 АНАЛИЗ, РАЗРАБОТКА ЭСКИЗНОГО И ТЕХНИЧЕСКОГО ПРОЕКТА ПРОГРАММЫ**

## **1.1 Анализ, информационной структуры предприятия и определение основной группы проблем**

В результате анализа программно информационной структуры организации были выявлены основные проблемы сотрудников предприятия, определены используемые основные программные средства и операционные системы.

В организации установлено 102 вычислительные машины типа стационарный компьютер и ноутбук, используемые в качестве рабочих станций, компьютеров учащихся и серверов. Расположенных в 45 кабинетах трех этажного здания типичного для общеобразовательных школ.

В основном парк рабочих станций основан на линейке операционных систем семейства Windows, с наличием нескольких серверов на основе операционной системы Linux. Так же присутствует сервер на основе Windows Server 2012r2. На нем развернуты службы WSUS, DHCP, DNS, Radius, центр распространения Kaspersky Security Center, центр сертификации, глобальный каталог контроллера доменов Windows, «лес» school20.local, групповые политики и службы удаленного доступа. В сети присутствует конвент фильтр на основе Dansguardian и Squid для фильтрации интернет ресурсов HTTP и HTTPS. Доступ к ресурсам в сети интернет осуществляется по «белому» списку.

Рабочие станции входят в домен school20.local, проходят аутентификацию посредством учетных записей доменной сети. Получают централизованные обновления операционной системы, установку программного обеспечения, групповых политик, обновлений антивируса непосредственно с контроллера доменов «леса» school20.local. На рабочих станциях установлены пакеты офисных программ, программы для обучения

детей, кодеки для просмотра видео и браузер для работы с системой «Сетевой город».

Основные проблемы: мелкие неисправности программного обеспечения, ошибки пользователей, неверное понимание принципов работы программного обеспечения и неумение работать с персональным компьютером. На основе этих данных можем составить типы основных заявок:

1. Консультации по работе с установленным программным обеспечением
2. Консультации по работе с оборудованием, цифровыми устройствами
3. Внесение изменений в настройки конфигураций программного обеспечения
4. Обновление установленного программного обеспечения (системного, пользовательского)
5. Установка нового программного обеспечения, добавление функционала, установка драйверов
6. Предоставить доступ пользователю к заблокированному интернет ресурсу
7. Иные мелкие вопросы.

Проведя анализ затрат времени на решение наиболее часто-встречающихся проблем приходим к выводу: среднее время на решение задач не превышает 10 минут. В следствии со спецификой работы образовательного учреждения (уроки, короткие перемены), доступ в кабинеты для решения данных проблем без приостановки или нарушения учебного процесса является практически невозможным. Соответственно решение поставленных, на специалиста ИТ службы, задач и физическое пребывание в месте неисправности накладывает большие временные ограничения, очень сильно замедляя рабочий процесс системного администратора. Исходя из этого, было принято решение разработать

программу, объединяющую и упрощающую работу системного администратора, позволяя решать мелкие проблемы практически в любое время и не нарушая учебный процесс. Принцип действия программы – удаленное подключение к рабочей станции, внесение изменения через командную строку или непосредственно взяв управление компьютером пользователя. С возможностью обмена сообщениями от пользователя к обслуживающему ИТ администратору, чтобы максимально оперативно уведомить специалиста о наличии проблем, а так же избавить администратора от необходимости тратить время на обход кабинетов. Так же необходимо создать обратную систему сообщений от системного администратора – пользователю, для своевременного информирования пользователя о том, что его проблема решена или не может быть решена или для других целей передачи информации непосредственно на компьютер пользователя. Программа должна содержать базу данных рабочих станций, и пользователей которые за них отвечают, чтобы максимально ускорить процесс решения поставленных на системного администратора задач по поддержке пользователей в организации.

Общая характеристика предприятия. Муниципальное бюджетное общеобразовательное учреждение средняя общеобразовательная школа №20 - это образовательное учреждение, в котором обучающиеся смогут получить: начальное, основное и среднее (полное) общее образование; возможность формирования развитой, творческой личности, способной активно адаптироваться в окружающем социуме; возможность постоянного физического совершенствования, оздоровления и коррекции состояния здоровья под наблюдением специалистов; возможность получения образования в соответствии с требованием социального заказа.

Дата создания образовательной организации: 1 сентября 1988 года.

Учредитель образовательного учреждения: Муниципальное образование город Нижний Тагил.



Функции и полномочия учредителя осуществляет: Управление образования Администрации города Нижний Тагил

Действует на основании устава от 22.05.2017г. Лицензия №14766 от 12.09.2011, срок действия бессрочная. Директор Уткина Марина Игоревна.

## 1.2 Назначение и область применения

Назначение разрабатываемой системы – управление информационными системами средней общеобразовательной школы №20. Предоставляющая возможность легко, быстро и удобно получить доступ к персональным компьютерам сотрудников организации. Автоматизируя основные, повседневные задачи при работе с информационными системами, изменяя параметры операционных систем, настроек приложений пользователей, исправление ошибок, установку обновлений, выполняемое удаленно, посредством стандартных протоколов управления, службами операционной системы Windows. Данная система должна является универсальной и предназначена для использования в средних и крупных образовательных учреждениях, а так же на других предприятиях с количеством информационных систем не менее 20. На основе операционных систем Windows, не ниже Windows XP SP3 и операционных системах Linux. Необходимо наличие доменной сети предприятия для использования возможности аутентификации администратора посредством пары логин/пароль учетной записи группы «администратор домена». Данное приложение так же может использоваться для обучения начинающего специалиста IT службы организации, получения начальных знаний по работе с программами удаленного администрирования. Показывая начинающему специалисту доступные инструменты управления информационными системами. Такие как удаленная консоль или PowerShell, функционал которой позволяет производить точную настройку системных параметров операционной системы. Помочь начинающему специалисту на практике проверить знания консолей управления. Помимо этого приложение может

быть использовано для обучения сотрудников организации при работе с другим программным обеспечением. Например, удаленно показать пользователю какую функцию программы выбрать для решения поставленной на пользователя задачи, или просто помочь пользователю. Помимо этого функционал программы позволит, обменивается сообщениями между пользователем и администратором. Посредством отправки сообщений через Веб-форму.

### 1.3 Технические характеристики. Постановка задачи. Спецификация технических и программных средств

Для создания программы управления информационными системами (далее ИС) предприятия необходимо определить список требований.

Для создания программы (приложения) требуется:

1. Определить среду/язык разработки программы.
2. Определить функционал программы, временные и постоянные переменные.
3. Определить необходимые внешние и внутренние компоненты, входящие в состав программы.
4. Разработать базу данных ИС (персональных компьютеров, серверов) соответствующей следующим характеристикам: определение расположения ИС в организации (этаж, № кабинета), сетевое (доменное) имя ИС, установленная операционная система на ИС, ответственного сотрудника ИС, дату установки ИС в место расположения.
5. Реализовать возможность добавления/редактирование записей базы данных непосредственно из программы.
6. Реализовать возможность сохранения, регистрационных данных пользователя, служебной информации приложения в файле конфигурации.
7. Обеспечить безопасность хранения конфиденциальной информации (данные аутентификации: для служб обмена сообщениями, администраторов доменных сетей, администраторов Linux и д.р.), файла

конфигурации используя динамическое шифрование-дешифрование посредством ключа.

8. Предусмотреть возможность безопасного входа в приложение посредством ввода пароля.

9. Предоставить возможность менять пароль пользователя для доступа к программе, а так же возможность изменять другие регистрационные данные (данные аутентификации: для служб обмена сообщениями, администраторов доменных сетей, администраторов Linux).

10. Разработать интерфейс программы (простой, удобный, интуитивно понятный пользователю).

11. Разработать Веб-форму для отправки сообщений пользователей.

12. Реализовать возможность отправки сообщений от администратора пользователю непосредственно на ПК.

13. Разработать возможность получения сообщений от пользователя администратору непосредственно в окно приложения.

14. Организовать возможность навигации по текущим и предыдущим сообщениям

15. Создать справочную информацию по функционалу программы.

Проведя анализ наличия программного обеспечения с аналогичным набором функционала, базы данных, служб обмена сообщениями, средствами поиска в сети интернет. Было выявлено отсутствие программ со схожими функциями. Были обнаружены программы с частичной реализацией: Team Viewer, Ammyu Admin, Readmin, AnyDesk, TightVNC. Во всех отсутствует встроенная база данных информационных систем. Team Viewer, Ammyu Admin, Readmin являются платными программами. Функционал бесплатных аналогов не отвечает требованиям, объявленным в постановке задачи.

Спецификация состава технических и программных средств. На текущий момент среди персональных компьютеров доминируют операционные системы семейства Windows. А среди мобильных устройств

Android и Apple, соответственно, что тот, кто собирается разрабатывать программное обеспечение, стремится писать программы, которые будут ориентированы на эти операционные системы. Последнее время количество наборов инструментов для разработчика неуклонно растет, начиная от Delphi и C# до Swift от Apple. От программиста требовалось только выбрать нужный набор инструментов исходя из соображений наличия собственных знаний в языке программирования и наличия жизненно необходимого для решения поставленной задачи функционала для создания собственной программы.

Быстрое развитие цифровой электроники, а так же потребность в эффективных средствах разработки программного обеспечения привели к появлению систем программирования (набор инструментов разработчика), ориентированных на так называемую «быструю разработку». В основе систем инструментов разработки (RAD-систем, Rapid Application Development — среда быстрой разработки приложений) лежит технология визуального проектирования и программирования на основе событий (Alert), смысл которого заключается в том, что система разработки берет на себя львиную долю работы, оставляя за программистом работу по конструированию диалоговых окон и функций обработки событий.

Скорость разработки программ посредством RAD-систем, огромная. Для решения поставленной задачи посредством визуального программирования программисту требуется намного меньше времени, благодаря автоматизации процессов создания визуальных объектов программы.

Набор инструментов программирования – RAD Studio XE6 Delphi – весьма быстрый, удобный набор инструментов для разработки приложений по решению широкого спектра задач для Windows. Для работы разработанных программы необходим IBM совместимый компьютер с тактовой частотой процессора 144МГц и выше, оперативной памятью 16 Мб,

жесткий диск объемом не менее 500Мб. Исходя из доступности и удобстве для разработки приложений, будет использоваться RAD Studio XE6 Delphi.

#### 1.4 Набор инструментов разработчика Embarcadero RAD Studio XE6 Delphi: назначение и общее описание среды. Структура среды программирования

Embarcadero RAD Studio XE6 Delphi — результат развития языка Турбо Паскаль, объектно-ориентированный язык программирования, в котором реализована функция создания программ с диалоговыми окнами, с интерактивными элементами. В отличие от Турбо Паскаля имеется возможность доступа к метаданным классов (то есть к описанию классов и их членов) в компилируемом коде, также называемом интроспекцией.

Расширенная поддержка Android позволит разработчикам на Delphi и C++ использовать преимущества растущего рынка мобильных устройств, используя единую базу исходных кодов.

Простая перекомпиляция позволяет переносить приложение на iOS, Windows или Mac OS X со стандартным или специальным пользовательским интерфейсом без каких-либо изменений в программном коде. RAD Studio XE6 поддерживает разные версии Android, включая 4.4 KitKat.

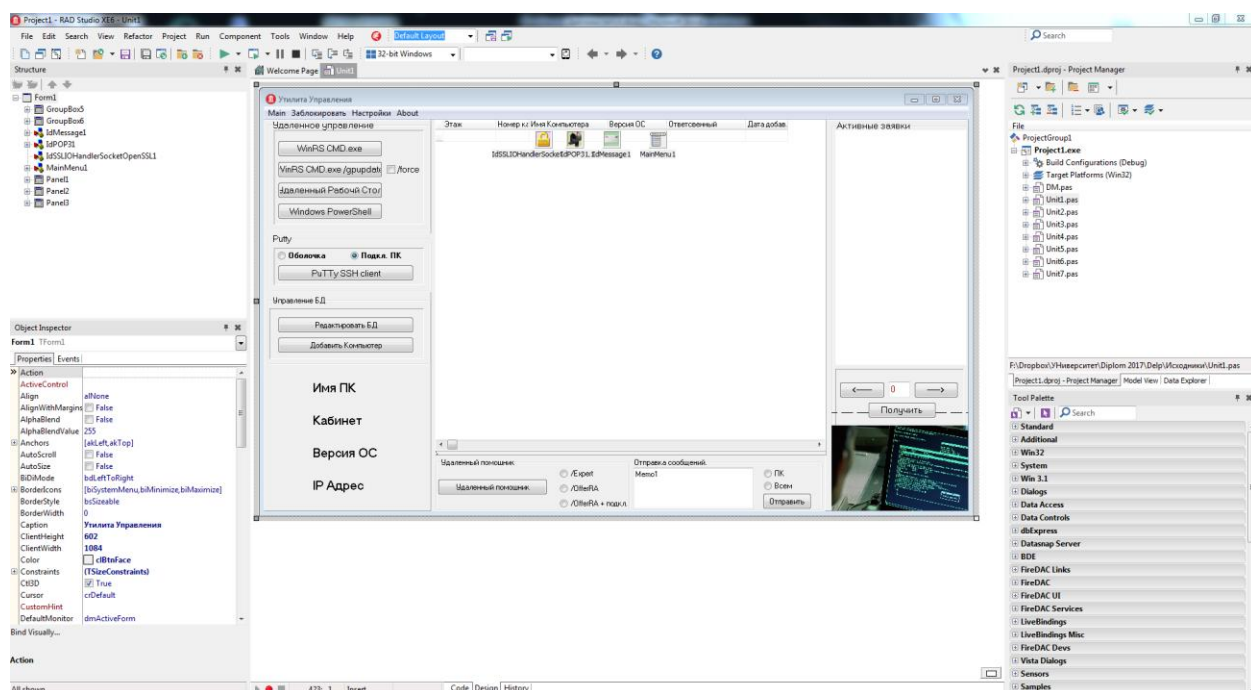
Новые компоненты для взаимодействия приложения (App Tethering) позволяют разработчикам дополнить имеющиеся Windows VCL приложения функционалом для мобильной работы без необходимости переноса приложения целиком. Разработчики могут создавать мобильные приложения как дополнения к существующим системам, разработанным под Windows.

Embarcadero RAD Studio XE6 упрощает использование всех преимуществ и стилей операционных систем Windows 7 или Windows 8.1, обеспечивая безболезненный переход от не поддерживаемой больше Windows XP к современным Windows 7 и Windows 8.1. При помощи RAD Studio XE6 разработчики могут модернизировать VCL приложения Windows: осуществляя взаимодействия многооконных приложений с использованием

новейших компонентов и, тем самым, придавая современный вид приложениям в стиле Windows 7 и Windows 8.1 или даже собственного дизайна; полностью управляя стилями внешнего представления ваших приложений, включая меню и границы окна. Кроме того, у разработчиков появилась возможность создать более производительные 64-битные варианты своих 32-битных приложений.

**RAD Studio XE6 Delphi.** Для реализации интерфейса Delphi использует библиотеку классов, в которой имеется огромное количество классов, поддерживающих форму и различные объекты формы (командные кнопки, поля редактирования и т. д.).

При создании и проектировании форм в программе RAD Studio XE6 Delphi, все необходимые объекты (программный код, зависимости, иные объекты среды разработки), автоматически добавляются к тексту программы.



*Рис 1. Пример интерфейса Набор инструментов Embarcadero RAD Studio XE6*

Структура RAD Studio XE6 Delphi программирования. Внешний вид среды программирования RAD Studio XE6 Delphi отчасти похож на другие

среды разработки (Рис 1) которые можно увидеть в Windows. К примеру, Borland C++, .NET, - это все MDI приложения и выглядят похоже на RAD Studio XE6 Delphi. MDI (Multiple Document Interface) - указывает специальный способ управления нескольких дочерних окон внутри одного окна.

Старые версии Delphi соответствуют другой спецификации, называемой Single Document Interface (SDI), и состоит из нескольких отдельно расположенных окон. Это было сделано из-за того, что SDI близок к той модели приложений, что используется в Windows 95.

Плюсы и минусы языка программирования Delphi и среды разработки RAD Studio XE6 Delphi. За скоростью, простотой и эффективностью Delphi следует ее популярность. Delphi имеет достаточно быстрый компилятор, создающий, очень неплохой объектно-ориентированный код. Помимо этого достоинством языка является его простота, по причине низкой сложности используемого в нем языка Паскаль. Наличие облегчающих разработку нововведений – в виде свойств (properties) программы, написанные на Delphi, не требуется снабжать дополнительными внешними библиотеками (в отличие от связки C++). В связи с чем, VCL показывает удобный, легко расширяемый объектно-ориентированный интерфейс к Windows, что ни как не мешает программисту разрабатывать приложения, углубляясь в изучение Windows API. Разработчикам собственных компонентов это приходится делать довольно часто. Модель программирования в RAD Studio XE6 Delphi - компонентная, что позволяет пользоваться программами, функциями и компонентами, написанными другими разработчиками, даже не зная, их исходного кода и не изучая его. В сети Интернет есть большое количество разработанных компонентов, львиная доля которых распространяется абсолютно бесплатно, что благополучно влияет на интерес к разработке сложных программных продуктов, содержащих большое количество элементов и библиотек сторонних разработчиков. Применение модели подключения внешних компонентов приводит к тому, что разработчику не

нужно тратить дополнительное время для разработки встроенных и дополнительных функций, которые уже были разработаны другими разработчиками. На ту работу, на которую в других случаях ушли бы недели, можно сделать за часы или даже минуты. К другим достоинствам можно отнести браузер классов, быстрый вывод подсказки, а так же авто завершение кода (code completion). Конечно, объектно-ориентированное программирование и использование заранее запрограммированных функций накладывает определенные ограничения, но для тех вещей, для которых ее писали, RAD Studio XE6 Delphi подходит практически оптимально. Из положительных нововведений, хотелось бы выделить свойства объектов (properties) и перегружаемость процедур и функций (overloading). Некоторые проблемы могут возникнуть при работе с функциями API. В частности может быть проблемой то, что основным языком для API все же является C++ или .NET, и на нем пишутся все новые Software Development Kit (SDK) и заголовочные файлы к ним. Исходя из этого, для работы с новыми SDK, необходимо разработать и написать код на Pascal, в котором определить интерфейсы данного SDK.

Минусы языка программирования Delphi и RAD Studio XE6 Delphi. Можно определить несколько основных недостатков. Главный, недостаток (и одновременно достоинство) - статическое присоединение (linking) библиотеки VCL и компонентов к исполняемому файлу. Необходимо сказать, что VCL можно присоединять и динамически, но тогда с каждым новым приложением придется распространять еще и VCL, а это дополнительный объем пространства, занимаемый при выполнении и хранении приложения. Но если не увлекаться наращиванием функционала на интерфейс и использовать в программе минимально необходимое число объектов, то исполняемый файл будет приемлемо невелик. Другой недостаток (и так же достоинство) основывается на том, что в используемой в RAD Studio XE6 Delphi парадигме формы (Forms) информация, такая как настройки компонентов, включая свойства, значения по умолчанию, хранится в exe-



файле, причем не оптимальным образом. Анализ VCL исходного кода демонстрирует, что при создании объектной формы происходит синтаксический разбор данных инициализации, что в свою очередь может её значительно замедлять. Еще один недостаток, который так же является достоинством, - это Object Pascal. Несмотря на относительную простоту, эффективность и легкость в изучении, ему не хватает части мощных средств C++. Например, отсутствуют шаблоны, перегрузки операторов и объектной модели, похожей на модель C++. Недостатком является то, что в RAD Studio XE6 Delphi имеется малое количество параметров оптимизации программного кода.

### 1.5 Базы данных и системы управления базами данных

База данных — представленная в объективной форме совокупность самостоятельных материалов (статей, расчётов, нормативных актов, судебных решений и иных подобных материалов), систематизированных таким образом, чтобы эти материалы могли быть найдены и обработаны с помощью электронной вычислительной машины (ЭВМ) [30 ].

Поясним определение, на примере базы данных образовательного учреждения (электронный журнал успеваемости учащегося). В ней присутствуют сведения об учащихся, родителях учащихся, адресе прописки и проживания, предметах, успеваемости, учителей и т.д. Доступ к этой базе должны иметь как сотрудники образовательного учреждения (Преподаватели, администрация и т.д.), так и учащиеся и их представители, которым необходимо знать свои отметки успеваемости, домашние задания, расписания занятий. При этом у каждого участника доступа к базе данных разные привилегии на просмотр и изменения. Например, учитель информатики может изменить оценки, пропуски учащихся класса, по дисциплине информатика, но не должен иметь доступ на изменение любой информации по другим предметам. Сам школьник должен иметь доступ к базе данных, в первую очередь, только на чтение, и только для его личной

успеваемости. Представители администрации образовательного учреждения должны иметь более широкие привилегии на доступ.

Поэтому эта, база данных должна содержать методы и средства, позволяющие определенному пользователю или группе пользователей оперировать только с теми данными, которые входят в его/их компетенцию. В результате взаимодействия данных, с методами, доступными конкретным сотрудникам учреждения, создается информация, которая используется на основании личных компетенций, производят ввод или редактирование данных в базе данных.

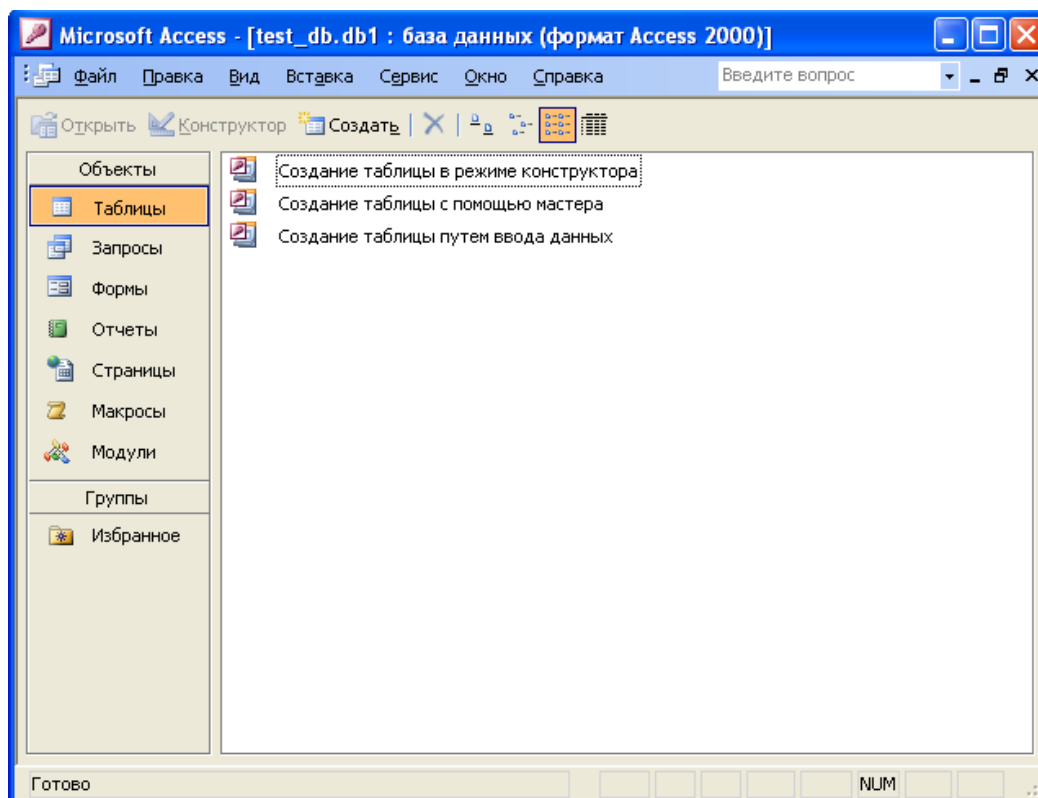
Говоря о базах данных, должны сказать и о понятии «системы управления базой данных» (СУБД). Это некий комплекс программных средств, разработанных и предназначенных для создания структуры базы, просмотр, наполнение ее содержимым, редактирование, обработки и визуализации информации. Визуализацией информации базы данных подразумевается отображение определенных данных в читаемом и удобном виде в соответствии с заданным критерием, а так же их упорядочение и последующая отправка на устройства вывода или передачи по каналам связи.

На текущий момент в информационной среде существует множество различных СУБД. Эти системы, могут по разному работать с разными объектами или предоставлять пользователю разный функционал и средства, по разному отображать (визуализировать) информацию в базе данных. Но всё же принципы работы опирается на часть основных устоявшихся понятий. Соответственно будет рассмотрена одна СУБД. Используя понятия свойственные именно ей. Но так же сможем получить общие понятия и знания на весь класс СУБД. Для описания возьмем популярную и известную СУБД Microsoft Access, входящую в пакет программ Microsoft Office.

Общие сведения баз данных на основе Microsoft Access. Microsoft Access 2007 является системой управления базами данных реляционного типа. Преимуществом баз данных на основе Microsoft Access является тот факт, что Access имеет достаточно простой и удобный графический

интерфейс, благодаря которому можно не только разработать и создавать собственную базу данных, но и разрабатывать приложения, используя встроенные средства разработки, в том числе средства разработки форм.

Отличительной особенностью Microsoft Access от других СУБД, в том, что Microsoft Access хранит все данные в едином файле (MDB – Microsoft Database), хотя по факту распределяет их по разным таблицам внутри одного файла, как и положено реляционной СУБД. К этим данным относятся не только данные в таблицах, но и иные объекты базы данных, которые будут описаны далее.



*Рис 2. Пример интерфейса Microsoft Access 2007 Мастера создания таблиц базы данных*

Для использования и разработки баз данных и приложений, почти всех основных функциональных операций в Microsoft Access предлагается большое количество Мастеров (Wizards) (Рис 2). Которые помогают разработчику, снимая основную нагрузку при работе с базой данных. Тем

самым снижая затраченное время, которое необходимо для создания приложений, благодаря возможности избежать рутинных действий.

Особенностью СУБД Microsoft Access 2007 является возможность создания многопользовательской СУБД Microsoft Access и получение одновременного доступа нескольких пользователей к общей базе данных посредством локальной вычислительной сети или на информационных системах, на которых возможна одновременная работа нескольких пользователей используя одну вычислительную станцию (несколько рабочих столов на одном персональном компьютере). Сеть обеспечивает обмен данными между компьютерами. Microsoft Access, управляет разграничением доступа к данным для разных пользователей СУБД, тем самым обеспечивая защиту данных при одновременной работе группы пользователей. Потому, что Microsoft Access не является клиент-серверной СУБД, её возможности по работе в многопользовательском режиме функционально ограничены. Для использования многопользовательского доступа к данным по сети с нескольких рабочих станций, выкладывают файл БД Access (с расширением mdb) на файловый сервер. При этом обработка информации из базы данных ведется на клиентском персональном компьютере – там, где запущено приложение, работающее с БД Access, исходя из особенностей организации файловых СУБД. Этот нюанс ограничивает использование Microsoft Access для обеспечения работы большого количества пользователей (больше 20 пользователей), а так же при большом количестве данных в таблицах базы данных, так как в связи с этим многократно возрастает нагрузка по передаче данных на вычислительную сеть предприятия.

Говоря о защите информации и разграничении доступа в Microsoft Access. Можно сказать об отсутствии надежных стандартных средств защиты. Стандартными способами защиты можно назвать – вход с использованием пароля к базе данных или пароля для доступа к элементам таблиц данных или с использованием паролей учетных записей

пользователей. Данный тип защиты не является надежным потому, что достаточно легко можно снять данный тип защиты.

Поддержка целостности данных в Microsoft Access соответствует только структуре базы данных небольшой или средней сложности. В ней отсутствуют: триггеры, хранимые процедуры, в виду этого, разработчикам необходимо возлагать поддержание целостности логики БД на клиентские программы.

Но при всех перечисленных и иных недостатках Microsoft Access обладает достаточно большим количеством преимуществ по сравнению с аналогичными системами, тем самым является популярной и часто используемой СУБД. Особенность распространенности среди пользователей связана не только с функциональностью и удобством, но и нацеленность СУБД на самую популярную линейку операционных систем Windows её разработчиком так же является Microsoft.

Microsoft, спроектировала и разработала множество программных продуктов совместимых между собой. Такие как операционные системы семейства Windows, офисные программы Office и многое другое. Программное обеспечение компании Microsoft является многоязычным и постоянно обновляется, получая новый функционал и улучшая существующий, в том числе и Microsoft Access. Позволяя улучшить совместимость баз данных Access с продуктами других разработчиков. В свою очередь разработчики RAD систем включают поддержку Microsoft Access в свои продукты «быстрой разработки программ». Что способствует быстрому распространению среди пользователей и разработчиков.

Microsoft Access предоставляет достаточное количество возможностей за сравнительно небольшую стоимость. В связи с этим отмечу особую ориентированность баз данных Microsoft Access на пользователя с разной степенью знаний в предметной области, что выражается в наличии большого количества средств помогающих пользователю создать необходимый продукт, обширную систему справки на разных языках, в том числе и на

русском, а так же понятный интерфейс. Все перечисленные средства помогают проектировать, создавать БД и выполнять выборку данных из нее.

Microsoft Access предоставляет пользователю диалоговые средства с разнообразным функционалом, которые позволяют решать поставленные задачи, одновременно не прибегая к разработке или изучению запросов на языках SQL или к программированию макросов-модулей на языке VisualBasicScript.

Microsoft Access позволяет выполнять задачи по импорту и экспорту данных в другие различные форматы для обеспечения наилучшей совместимости с другими программными продуктами. В том числе в таблицы Excel, текстовые файлы, или другую клиент-серверную СУБД через механизм драйвера ODBC.

Так же к преимуществу можно отнести то, что, в Microsoft Access развиты встроенные средства разработки приложений. Часть разработанных приложений, распространяемых среди пользователей, содержат код, разработанный на языке Visual Basic for Applications (VBA). Соответственно VBA является почти единственным полнофункциональным средством для выполнения многих стандартных задач присутствующих в Microsoft Access (создание линейки команд SQL, работа с переменными, обработка ошибок, использование Windows API и т. д.), для создания сложных приложений необходимо наличие знаний в области объектной модели Microsoft Access и VBA.

Кроме VBA, в средствах программирования Access используется язык макрокоманд. Программы, созданные на этом языке, называются макросами и позволяют связывать отдельные действия между собой, реализуемые с помощью форм, запросов или отчетов. Макросы запускаются событиями. Событие это одиночное или совокупность действий при диалоговой работе в форме или базе данных выполняемое пользователем или системными событиями (пример время и т.п.). Определенное событие вызывает определенный, заранее созданный макрос.

Получается что Microsoft Access, являясь по функционалу СУБД, одновременно предоставляет пользователю дополнительные возможности разработчика. Это не только удобная, гибкая и достаточно простая в использовании СУБД, но и достаточно производительная и функциональная система для разработки приложений работающих с базами данных.

#### 1.6 Разработка HTTP Web-страницы на основе HTML и PHP5, описание языка web-программирования. Описание Sendmail

HTTP – основной протокол, разработанный для передачи данных посредством web страниц. Данный протокол широко распространен в сети интернет. Благодаря ему имеется возможность: передавать гипертекстовые документы. Содержащие ссылки на другие ресурсы, мультимедиа контент (картинки, музыка, видео), скрипты, программы и другие пассивные и интерактивные объекты.

Язык HTML, достаточно прост для понимания, с достаточно строгой, но в то же время адаптивной структурой документа. HTML, состоит из набора разных тегов, элементов разметки документов. Содержащие объекты, такие как текст, картинки, видео-аудио, скрипты и другие программные объекты, их параметров, определяющих свойства и параметры данного объекта.

HTML имеет определенный синтаксис тегов, благодаря которому программа просмотра документа (Веб-браузер) отображает содержимое страницы. Отображая картинки, видео таблицы и другие мультимедиа элементы на тех местах, на которых это было задумано программистом. При этом сами теги, параметры и свойства тегов, программой просмотра Веб документа не отображаются. HTML поддерживает механизмы гипертекстовых ссылок, которые позволяют пользователю, благодаря программе просмотра Веб документов, без особых сложностей производить навигацию, как по текущему документу, так и быть перенаправленным на другой Веб-документ, на этом или любом другом Веб-сайте.

Язык HTML постоянно совершенствуется и обновляется, продолжая тем самым развиваться. HTML, скорее всего, будут использоваться, и развиваться в дальнейшем. Познавая HTML глубже, Веб-программисты имеют возможность создавать Веб-страницы, которые в реальном времени могут быть получены и просмотрены многими современными Web-браузерами, как сейчас, так и в будущем. Работа по HTML - это способ усвоить особенности создания Веб-документов на стандартном языке, используя расширения, такие как PHP.

PHP - язык сценариев, который стал намного выше своего названия. PHP - это аббревиатура от слов Personal Home Page, которая переводится как персональная домашняя страница. Первая версия набора инструментов PHP была разработана Расмусом Лердорфом в 1994 г. и представляла собой набор скриптов для отслеживания посетителей Веб-страницы. Со временем PHP из набора инструментов превратился в функциональный язык Веб-программирования, а его наименование было изменено как рекурсивное образование PHP HyperText Preprocessor (препроцессор гипертекста PHP).

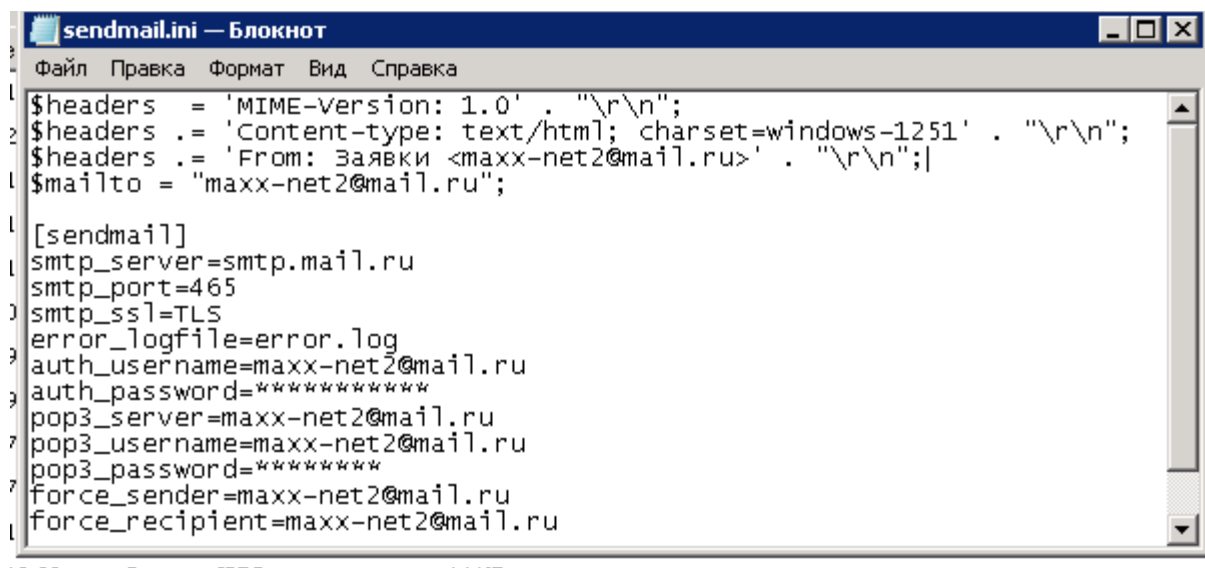
В настоящее время PHP - это серверный язык создания сценариев. Сценарии и скрипты написанные на PHP, интегрированные в структуру Веб-документа, созданного на основе языка HTML, выполняются серверном при посещении страницы пользователем. Результатом данной обработки информация HTML-текст передается браузеру или техническая информация службам сервера.

Самое удобное в PHP то, что это Веб-ориентированный язык. Язык, специально предназначенный для использования в интернете, освоив PHP язык, будь то пользователь или разработчик, сможет решить множество поставленных задач в области создания страниц сайта, аналитических и других структур, которые не возможно или не продуктивно решать на языке HTML.



Основным достоинств РНР является то, что его можно внедрить в HTML код. РНР код можно внедрить непосредственно в HTML, или наоборот HTML можно внедрят в РНР код, что и делает его перспективным и необходимым языком.

Описание Sendmail. Для отправки электронных сообщений из Веб-сайта необходим набор библиотек, скриптов и программ выполняющих создание, подготовку, отправку сообщения по средствам сети интернет с использованием безопасного канала (шифрование). Sendmail — один из старейших агентов передачи почты (MTA — mail transfer agent). Распространяется бесплатно вместе с исходными кодами. Существуют версии программы для практически всех операционных систем и аппаратных платформ. В данное время разрабатывается фирмой Sendmail Inc. Данный MTA агент отправляет почту в два этапа — почтовые сообщения собираются в очередь, а на втором этапе данные непосредственно отправляются. Сообщение состоит из следующих основных частей: конверта, заголовка и тела сообщения. Конверт состоит из адреса отправителя, адреса получателя и информации рассылки, которая используется программами подготовки, рассылки и получения почты. Конверт остается невидимым для отправителя и получателя почтового сообщения. При создании сообщения Веб-сайт должен передать агенту сообщение с определенными параметрами, которые содержат адреса, информацию о рассылке и другие данные (Рис 3).



```
1 $headers = 'MIME-version: 1.0' . "\r\n";
2 $headers .= 'Content-type: text/html; charset=windows-1251' . "\r\n";
3 $headers .= 'From: Заявки <maxx-net2@mail.ru>' . "\r\n";|
4 $mailto = "maxx-net2@mail.ru";

5
6 [sendmail]
7 smtp_server=smtp.mail.ru
8 smtp_port=465
9 smtp_ssl=TLS
10 error_logfile=error.log
11 auth_username=maxx-net2@mail.ru
12 auth_password=*****
13 pop3_server=maxx-net2@mail.ru
14 pop3_username=maxx-net2@mail.ru
15 pop3_password=*****
16 force_sender=maxx-net2@mail.ru
17 force_recipient=maxx-net2@mail.ru
```

*Рис 3. Данные передаваемые скриптом из web-формы при отправке сообщения в Sendmail*

## 1.7 Шифрование и дешифрование конфиденциальных данных посредством ключа

Одной из поставленных задач в п.1.3 является требование сохранения программой служебной информации и данных аутентификации: для служб обмена сообщениями, администраторов доменных сетей, администраторов Linux и их последующее шифрование. Необходимо определить способ шифрования, алгоритм, и способы дешифровки зашифрованных данных. Рассмотрим популярные методы шифрования.

Один из распространенных шифров ROTate или шифр Цезаря. Он известен многим, в связи со своей распространенностью. Называется так, потому что его использовал сам Юлий Цезарь. На самом деле шифр Цезаря — это не один шифр, а несколько, использующих один и тот же принцип. Получателю нужно сказать, какой из ключей используется. Особенностью данного шифра является его простота. Для того чтобы зашифровать текст необходим ключ — число. Ключом является значение, на сколько букв относительно алфавита сместится символ. Например, для ключа 2 каждая буква смещается на 2 позиции по отношению к алфавиту. Для английского

языка, А заменяется на С, В на D, и т.д. «ROT1» значит «ROTate 1 letter forward through the alphabet» (англ. «сдвиньте алфавит на одну букву вперед»).

Другим достаточно известным способом шифрования является Транспозиция. В транспозирующих шифрах буквы переставляются по заранее определенному правилу. Например, если каждое слово пишется задом наперед, то из «all the better to see you with» получается «lla eht retteb ot ees joy htiw». Другой пример — менять местами каждые две буквы. Таким образом, предыдущее сообщение станет «la tl eh eb tt re ot es ye uo iw ht». Подобные шифры использовались в Первую Мировую и Американскую Гражданскую Войну, чтобы посылать важные сообщения. Сложные ключи могут сделать такой шифр на первый взгляд довольно трудным для дешифровки, но многие сообщения, закодированные подобным образом, могут быть расшифрованы простым перебором ключей на компьютере. Воспользуемся этими методами и усилим их таблицей значений кодировки Unicode, воспользуемся функцией XOR к ключу для шифрования персональных данных.

В таблицах кодировок каждому символу алфавита плюс дополнительным символам соответствует определённое целое число в двоичной системе, с помощью двоичного кода можно закодировать текстовую информацию. Восемь бит информации достаточно для кодирования 256 различных символов. Этого хватит, чтобы выразить различными комбинациями все символы английского и русского языков, как строчные, так и прописные, а также знаки препинания, символы основных арифметических действий и некоторые общепринятые специальные (дополнительные) символы.

Если брать во внимание ASCII (American Standard Code for Information Interchange – стандартный код информационного обмена). В системе ASCII закреплены две таблицы кодирования базовая и расширенная. Базовая

таблица закрепляет значения кодов от 0 до 127, а расширенная относится к символам с номерами от 128 до 255.

Первые 32 кода базовой таблицы, начиная с нулевого, отданы производителям аппаратных средств. В этой области размещаются управляющие коды, которым не соответствуют ни какие символы языков. Начиная с 32 по 127 код, размещены коды символов английского алфавита, знаков препинания, арифметических действий и некоторых вспомогательных символов.

Исходя из всего изложенного, логично будет написать программный код, сдвигающий номер символа посредством команды XOR относительно ключа и статического математического действия, а так же необходимо учесть, что при шифровании результирующий код не может быть меньше 32 и больше 255 для таблицы ASCII, либо больше 65 536 для таблицы Unicode. Алгоритм должен смешать номер символа на значения 32 и учитывать длину таблицы кодировки. Так же полученное значение должно сохраняться, и храниться отдельно от кода программы. Так же алгоритм должен поддерживать дешифровку, процедуру обратную шифрованию. С использованием того же цифрового ключа, который использовался при шифровании. А так же учитывать минимальную (номер символа не меньше 32) и максимальную длину таблицы кодировки. Помимо этого алгоритм должен позволять получать разные значения ключа из разных форм или других алгоритмов. Для возможности шифровать разные данные разными ключами.

## **2 РАЗРАБОТКА КОМПОНЕНТОВ СИСТЕМЫ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ ИТ СЛУЖБЫ ПРЕДПРИЯТИЯ**

### **2.1 Разработка базы данных**

На текущий момент времени существует большое количество программных средств и СУБД, позволяющих создавать и использовать локальные (dBASE, FoxPro, Access, Paradox) и удаленные (Interbase, Oracle, Sysbase, Infomix, Microsoft SQL Server) базы данных.

В состав Embarcadero RAD Studio XE6 входят компоненты, позволяющие разрабатывать программные продукты с поддержкой работы с файлами данных, созданными на основе Microsoft Access.

Необходимо определить схему данных и перечень таблиц. Для хранения в базе данных информации об информационных системах общеобразовательной школы достаточно создать одну таблицу содержащие следующие поля:

1. Код (тип данных: счетчик, возможные значения: длинное целое)
2. Этаж (тип данных: текст, размер поля: 6, тип источника: несколько значений, обязательное поле: да);
3. Номер кабинета (тип данных число, возможные значения: целое, обязательное поле: нет);
4. Имя компьютера (тип данных: текст, размер поля: 25, обязательное поле: да);
5. Ответственный (тип данных: текст, размер поля: 30 обязательное поле: нет);
6. Версия ОС (тип данных: текст, размер поля: 15, тип источника: несколько значений, обязательное поле: нет);
7. Дата добавления (тип данных: дата, формат поля: длинный формат даты, обязательное поле: нет).

В соответствии с описанными требованиями была разработана и создана база данных, в формате MDB.(Рис 4).

Код	Этаж	Номер каби	Имя Компь	Ответственн	Версия ОС	Дата добав.
5	1 Этаж	103	pc-103xp	Иванов	Windows XP	04.05.2016
6	1 Этаж	104	pc-104xp	Петров	Windows XP	13.04.2016
7	1 Этаж	105	pc-105xp	Сидоров	Windows XP	10.03.2016
8	1 Этаж	106	pc-106xp	Сергеев	WindowsVista	07.01.2016
9	1 Этаж	107	pc-107xp	Биймурзаева	Windows XP	15.10.2015
10	1 Этаж	108	pc-108xp	Быстрова	Windows XP	27.06.2015
11	2 Этаж	201	pc-201w7	Белова	Windows XP	28.10.2015
12	2 Этаж	202	pc-202w7	Сидорова	WindowsVista	19.06.2015
13	1 Этаж	203	KOLBASOIDh	Захаров	Windows 7	29.06.2015
14	1 Этаж	111	home	Терещенко	Windows XP	11.12.2015
15	3 Этаж	314	pc-314t	Глушков	Windows XP	06.05.2016
16	2 Этаж	209	pc-209k	Иванченко	Windows 7	19.12.2015
17	2 Этаж	208	pc-208w7	Попова	Windows 7	27.04.2016
20	1 Этаж	101	dc1-school20	Задорин	Windows 8	08.04.2016
21	1 Этаж	1	KOLBASOIDhoi		Windows 7	10.12.2015
*	(№)	1	NoName			

Рис 4. Базы данных Microsoft Access

#### 2.1.1. Web-форма отправки сообщений-заявок

Для удобной и доступной системы отправки заявок пользователей создадим Web-форму отправки сообщений администратору.

Для создания формы будет использоваться язык Web-программирования php, html и скрипт генерации, и отправки сообщений, посредством электронной почты. Для начала создадим схематический вид интерфейса пользователя. (Рис.5)

Интерфейс должен обладать.

1. Простотой, для легкой ориентации и удобства пользователя.
2. Содержать поля для ввода данных.
  - 2.1. Поле ввода имени пользователя. Узкое поле длиной не менее 60 символов. Тип вводимых значений: текст.
  - 2.2. Поле ввода кабинета пользователя заявителя. Узкое поле длиной не менее 20 символов. Тип вводимых значений: числовой.

2.3. Поле ввода сведений о проблеме пользователя. Основная форма. Размер не менее 5 строк. Тип вводимых значений: текст.

2.4. Динамически сгенерированный Объект-картинка с цифровыми значениями. Для защиты от роботов и вирусной рассылки сообщений.

The diagram shows a web form template enclosed in a rectangular border. It contains the following elements from top to bottom: a text input field labeled 'Имя', another text input field labeled 'Кабинет', a large text area labeled 'Текст1', a small rectangular box containing the numeric string '1234567890', and a button labeled 'Отправить'.

*Рис 5. Шаблон Веб-формы для отправки информационных сообщений*

2.5. Кнопка отправки сообщения. Кнопка отправки введенных сведений в скрипт для генерации и отправки e-mail сообщения для администратора.

Определим необходимые компоненты для создания формы и запуска сайта.

1. Web Сервер IIS версии 7.5
2. PHP Версии 5.4.13
3. Скрипт sendmail версии 32
4. Библиотеки SSL-TLS ssleay32.dll, libeay32.dll.

### 2.1.2 Временные и постоянные переменные

В программе используются следующие переменные, в которых хранятся входные и выходные данные:

#### Временные

- PCName:string = '0'; Имя компьютера из БД.
- Cabinet:string = '0'; Кабинет из БД.
- OS:string = '0'; Версия операционной системы из БД.
- WindowsName, WindowsPass, LinuxName, LinuxPass:string;

регистрационные данные по типу логин/пароль хранящиеся в файле конфигурации

#### Постоянные

- WINSOCK\_VERSION = \$0101;. Версия модуля работы с ЛВС.

### 2.1.3 Внешние и внутренние компоненты, Файл конфигурации, шифрование паролей пользователя

Для реализации данного проекта будут использоваться следующие модули:

#### внутренне

- DM - Модуль работы с БД Access
- IniFiles – модуль работы с файлами конфигурации .ini
- WinSock – модуль для работы с ЛВС
- Unit2-Unit7 – визуальные формы программы.

#### Внешние

- cmd.exe – командная строка Windows
- winrs – консоль для работы с удаленными ИС
- RDP - клиент – приложение для управления удаленными

рабочими столами

- Putty - клиент универсальный консольный ssh/telnet клиент
- PowerShell – Windows совместимый клиент управления ИС на

основе службы MS management.



- Winsent – программа отправки сообщений пользователям.

Файл конфигурации, шифрование паролей пользователя

Для хранения параметров будем использовать следующие переменные

1. WindowsName – имя пользователя Windows
2. WindowsPass – Пароль Windows
3. LinuxName - имя пользователя Linux
4. LinuxPass - Пароль Linux
5. POP3Server – POP3 почтовый сервер
6. POP3Port – POP3 порт
7. LoginPOP3 – имя пользователя электронной почты
8. PasswordPOP3 – пароль пользователя электронной почты
9. Password на доступ к программе
10. SSLTLS – выбор одного из 5 типов шифрования при

подключении к POP3.

Данные хранимые в переменных под пунктами 1-10 невозможно хранить в скомпилированном виде и иметь при этом возможность изменить, а так же сохранять их во время работы с приложением. Было принято решение хранить параметры программы в отдельном внешнем файле – файле конфигурации. Расположение файла в каталоге с программой ./cfg.ini, тип ini текстовый файл в кодировке UNICODE.

Параметры: WindowsName, WindowsPass, LinuxName, LinuxPass, LoginPOP3, PasswordPOP3, Password являются конфиденциальными. И в целях безопасности не могут храниться в открытом виде. По этой причине было принято решение применить для данных параметров шифрование с закрытым персональным ключом (Рис 6). Для каждого параметра применяется одна функция шифрования, но с подачей разных ключей. Благодаря этому, компрометация одного из ключей не повлечет за собой компрометацию всех сохраненных паролей. При проверке или отправки паролей в другие формы и приложения используется обратная функция – дешифровки, с применением тех же персональных ключей шифрования.

Ключи для каждого из перечисленных параметров, сохраненных в файле конфигурации, хранятся в коде программы [Приложение 3].

Тип шифрования – посимвольное кодирование, получение порядкового значения символа пароля, выполнение математических операций с этим значением с применением значений ключа. Получение нового порядкового значения, получение из него нового символа и запись его в файл.

```
function Crypt(s:string;code:boolean):string;
var
    s: входные данные
    i, Delta, Res: integer;
begin
    Result:='';
    for i:=1 to Length(s) do
    begin
        Delta:=((i xor Pas) mod (256-32));
        if code then      если code=1; тогда шифруем
            Res:=((ord(s[i])+Delta) mod (256-32))+32
        else
            begin      если code=0; тогда дешифруем
                Res:=ord(s[i])-Delta-32;
                if Res<32 then
                    Res:=Res+256-32;
            end;
            Result:=Result+chr(Res);
        end;
    end;
end;
```

Рис 6. Пример кода шифрования - дешифровки

#### 2.1.4 Интерфейс программы RemoteNetworkTracer

В связи со спецификой программы, а именно основной функционал это удаленное администрирование. Весь проект и разработанная программа будет иметь имя «Remote Network Tracer» (Удаленный сетевой наблюдатель).

Макет интерфейса программы разработан в соответствии с требованиями в программе MS Paint.

А. Основная форма (Рис 7).

A1. Центральная динамически обновляемая форма. Для отображения и возможности взаимодействия с элементами базы данных. Предоставляющая возможность выбрать элемент и получить актуальную информацию о ней в поле №6

A2. Блок формы с клавишами. Кнопка «Редактирование БД»-открывает дополнительную форму (Рис 8). Кнопка «добавить ПК» позволяет добавлять новые элементы базы данных

A3. Блоки с функциональными клавишами удаленного управления. WinRS, cmd, удаленный рабочий стол, PowerShell. Putty, удаленный помощник.

A4. Блок форма для отображения актуальных заявок от пользователя, динамически обновляемая. С возможностью навигации в блоке управления Веб формой.

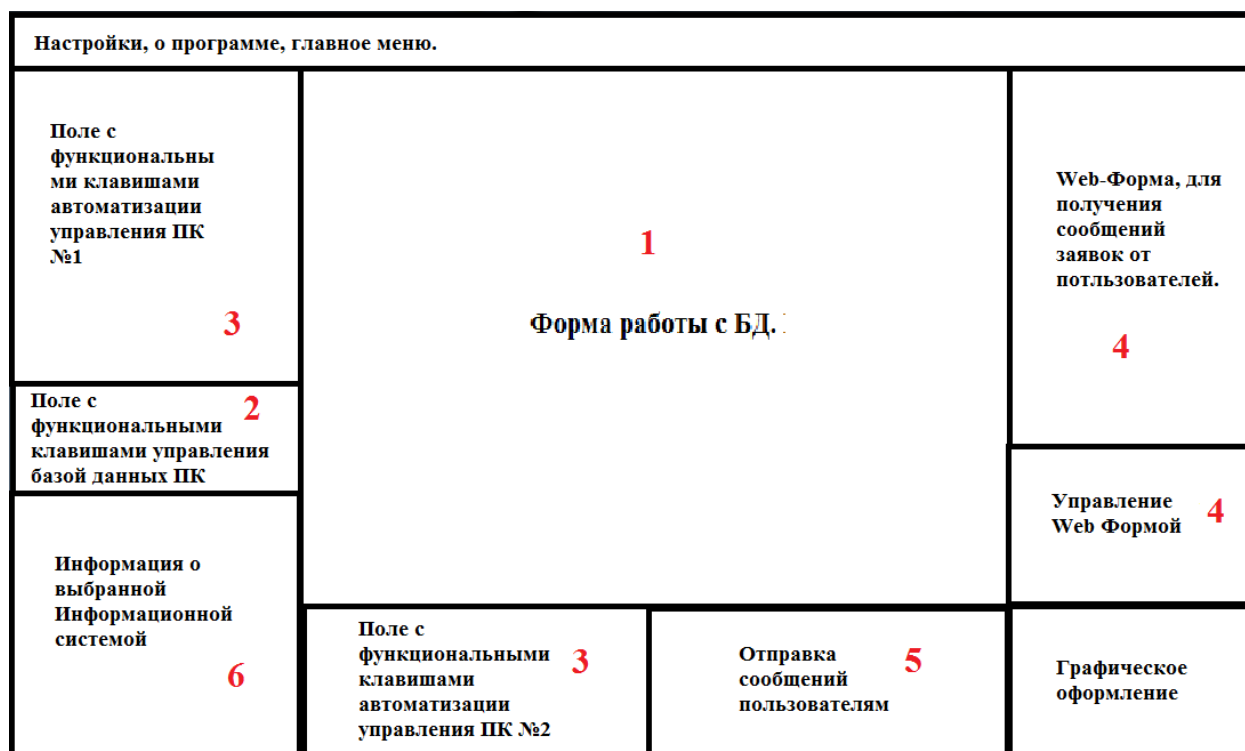


Рис 7. Макет интерфейса программы

A5. Блок форма отправки сообщений компьютерам пользователей.

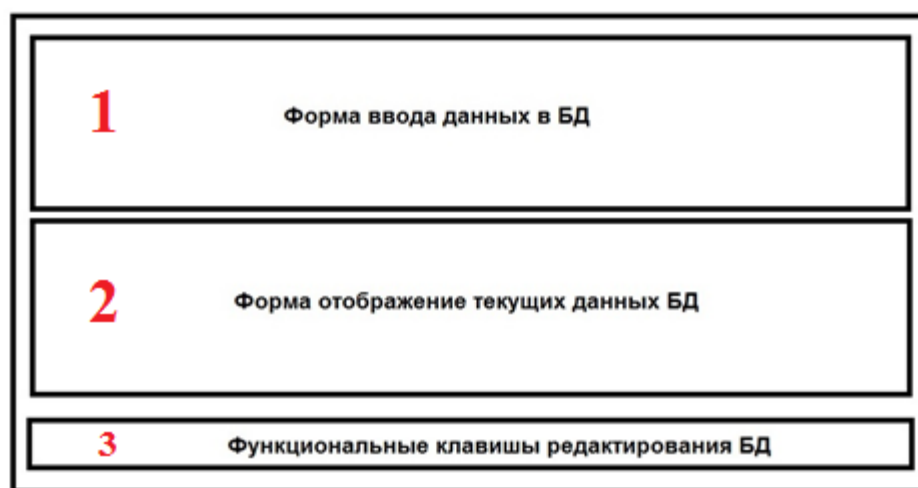
А6. Блок форма для отображения актуальной информации о выбранной ИС.

Б. Дополнительная Форма, предназначенная для добавления или изменения элементов базы данных (рис 8).

Б1. Блок форма с элементами для ввода значений элементов выделенного значения базы данных

Б2. Динамически обновляемая блок форма. Отображающая элементы базы данных.

Б3. Блок функциональных клавиш для навигации, добавления, сохранения текущего состояния БД.



*Рис 8. Макет Формы редактирования БД*

## 2.2 Спецификация программы RemoteNetworkTracer

Программа объединяет базу данных, в которой хранится информация об информационных системах, внешние компоненты управления персональными компьютерами учреждения и программный код, обеспечивающий их взаимодействие.

В программе объединены следующие подсистемы: системы отправки сообщений пользователь-администратор и администратор-пользователь, инструменты управления операционными системами семейства Windows и

Linux. С ведением общей базы данных персональных компьютеров организации, с указанием их размещения, сетевого имени, ФИО сотрудника-пользователя персонального компьютера, тип операционной системы. Программа умеет работать с базой данных на основе Microsoft Access. Поддерживая функции: Добавление/редактирование/удаление элементов базы данных, получение динамического IP-Адреса персонального компьютера из локальной сети и передача этих данных другим элементам программы. Добавлена возможность шифрования регистрационных данных пользователя ключом, с возможностью дешифровки, посредством передачи ключа алгоритму шифрования. Реализована функция блокировки входа в приложение посредством пароля. Для повышения безопасности работы с приложением.

При работе с базой данных программный код использует сведения из выбранного элемента таблицы, передает их внешнему компоненту вместе с сохраненными регистрационными данными пользователя для проведения аутентификации.

Элементами Базы данных является набор однородной, как правило, упорядоченной по некоторому критерию, информации.

Следует обратить внимание, что каждая запись состоит из одинаковых полей. Некоторые поля могут быть не заполнены, однако они все равно присутствуют в записи. Каждая строка таблицы соответствует записи, а ячейка таблицы — полю. При этом заголовок столбца таблицы — это имя поля, а номер строки таблицы — номер записи. Информация компьютерной базы данных выводится на экран в виде таблицы.

Таблицы базы данных физически хранятся в отдельном файле. В некоторых случаях информацией для программы, работающей с базой данных, может быть вся таблица. Однако, для выполнения основного функционала программе необходима не вся информация, находящаяся в базе данных, а только определенная ее часть. Остальные данные являются информационными и предназначены для специалиста IT службы

учреждения. Программа посредством запроса выбирает необходимые ячейки из строки для дальнейшей обработки. А именно берет из выбранной строки в форме необходимую в данном случае информацию и присваивает её переменным. В дальнейшем одна часть информации выводится в окно программы как информация, другая часть вставляется в код функциональных кнопок. Позволяя выполнять функции ИТ администрирования без ручного ввода и передачи команд системным приложениям.

### 2.3 Описание программы RemoteNetworkTracer

Программа написана в среде программирования Embarcadero RAD Studio XE6. Для запуска программы необходима операционная система Windows 98, NT, 2000, Me, XP, Vista, 7, 8, 8.1, 2012. Программа не предназначена для работы и не тестировалась в DOS, Linux, Android, MacOS.

Функциональное назначение. Программа является универсальной и может использоваться не только в образовательных учреждениях, но и на коммерческих предприятиях.

Функциональные клавиши.

I. WinRS cmd.exe – автоматическое подключение к выбранной ИС используя комбинацию логин/пароль для ОС Windows и запуск консоли CMD на удаленной ИС.

II. WinRS gpupdate(/force)- автоматическое подключение к выбранной ИС используя комбинацию логин/пароль для ОС Windows и получение новых/обновленных групповых политик. Ключ /force запускает принудительное обновление всех политик компьютера.

III. Удаленный рабочий стол – автоматическое подключение к выбранной ИС по протоколу RDP используя комбинацию логин/пароль для ОС Windows для управления удаленным рабочим столом пользователя.

IV. Putty SSH (функция Подключение к ПК) - автоматическое подключение к выбранной ИС по протоколу SSH используя комбинацию логин/пароль для ОС Linux и запуск удаленной консоли управления.

Функция «Оболочка» Запуск приложения Putty без подключения в режиме оболочки. Для ручного подключения к другим ИС посредством других протоколов и комбинаций логин/пароль.

V. PowerShell - автоматическое подключение к выбранной ИС используя комбинацию логин/пароль для ОС Windows используя систему управления PowerShell на удаленной ИС.

VI. Отправка сообщений кнопка «отправить» – использует бесплатное приложение winsent позволяющее отправлять сообщения посредством передачи приложению параметров, получателя и текст сообщения посредством командной строки. Отправка происходит посредством передачи имени компьютера из базы данных, передачи сообщения из формы и непосредственную отправку сообщения получателю.

VII. Кнопка «получить» и кнопки «<---» «--->» формы получения сообщений. Кнопка «получить» отображает последнее сообщение, отправленное через специальную форму от пользователей, позволяя благодаря кнопкам-стрелочкам переходить к следующим или предыдущим сообщениям.

VIII. Кнопка «удаленный помощник» позволяет подключиться в режиме удаленной помощи к ПК пользователю посредством стандартным клиент-серверным приложением удаленного помощника Windows (msra).

IX. Функции меню «настройки»

а. Настройка почты. Запуск формы с вводом настроек pop3 сервера, порта, типа шифрования, и данные для аутентификации пользователя на сервере электронной почты.

б. Настройка учетных данных. Запуск формы для ввода и сохранения пары логин/пароль для подключения к рабочим станциям для Windows и Linux.

с. Смена пароля. Запускает форму смены пароля на доступ к программе.

Х. Меню «заблокировать» блокирует доступ к приложению до ввода пароля пользователя.

Вызов и загрузка, входные и выходные данные. Используемые технические средства (минимальные требования). Минимальные *системные требования*: процессор Pentium-133 и выше, ОЗУ 16Мб, место на диске не меньше 2Мб.

Исполняемый файл программы – RemoNT.exe (Remote Network Tracer). В родительском каталоге программы также содержится файл Базы Данных – PC.mdb. Запустить программу можно пользуясь стандартным приложением для Windows «проводник» или через «Мой компьютер», просмотрев содержимое диска.

Входные данные представляют собой данные БД из файла PC.mdb и файла конфигурации cfg.ini.

Выходные данные выводятся на экран программы и присваиваются рабочим переменным, сохраняются в базу данных PC.mdb и в файл конфигурации cfg.ini.

## 2.4 Тестирование, Внедрение и функционал

Тестирование было проведено на примере Базы Данных в каталоге \BD\PC.mdb. В базу данных было произведено внесение данных о ИС. Всего было введено 20 компонентов. Все данные были введены в случайном порядке. Для начала произвольная ИС (ИС доступная посредством ЛВС на момент тестирования) была выбрана из списка формы работы с БД, далее посредством поочередного нажатия были проверены все функциональные клавиши. По такому же принципу были проверены форма редактирования БД и форма ввода логин/пароля. Программа не дала сбоев и ошибок. Таким же образом произвели тестирование на ИС не доступной посредством ЛВС на момент проверки. Программа успешно прошла тестирование.

Условия выполнения программы. На удаленной ИС должны быть настроены и запущены службы MS management и службы удаленных



рабочих столов. Параметры безопасности должны позволять Администратору с реквизитами указанными во вкладке «Логин/пароль» подключаться к перечисленным службам. В настройки фаервола должны быть внесены разрешающие правила для доступа из локальной сети на порты по протоколам TCP и UDP: 22, 23, 80, 135, 137, 138, 139, 443, 445, 3389,

На рабочие станции посредством групповых политик необходимо установить актуальные версии Windows PowerShell, WinRM, WinSent. Принудительно включены службы: удаленный доступ к рабочим станциям посредством Web Services for Management (WS-Management), удаленный помощник и клиент RDP.

## 2.6 Выполнение программы.

Для того чтобы запустить программу на панели управления щелкните кнопкой мыши кнопку Пуск. Выберите в развернувшемся меню пункт «Программы» - «проводник». В проводнике выберите нужный каталог с файлом RemoNT.exe и дважды щелкните по нему левой кнопкой мыши.

После запуска программы отобразится окно ввода пароля пользователя. Пароль хранится в зашифрованном виде в файле конфигурации, при удалении файла или удалении пароля вход в форму возможен только после ввода «специального» пароля разработчика. (Рис 9)

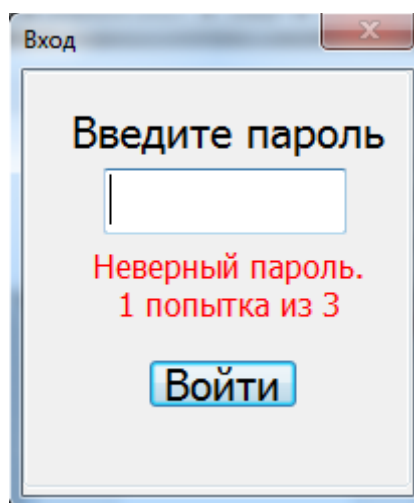


Рис 9. Форма входа в программу

Для ввода действительного пароля дается 3 попытки, после чего форма закрывается. После успешной процедуры аутентификации на экране будет отображено окно программы. (Рис 10)

Для выбора ИС из списка необходимо нажать на любую ячейку в строке необходимой ИС (в данном примере выделена строка кабинета 111). В левом информационном поле отобразится информация о данной ИС. Строка «ip адрес» автоматически отобразит ip(v4) ИС запросив его у локального DNS(если присутствует) или проведет прямой запрос. Во втором случае если ИС в данный момент недоступна, отобразится надпись «IP не определен». Если ИС не выбрана после запуска программы. По умолчанию выбирается первая ИС БД.

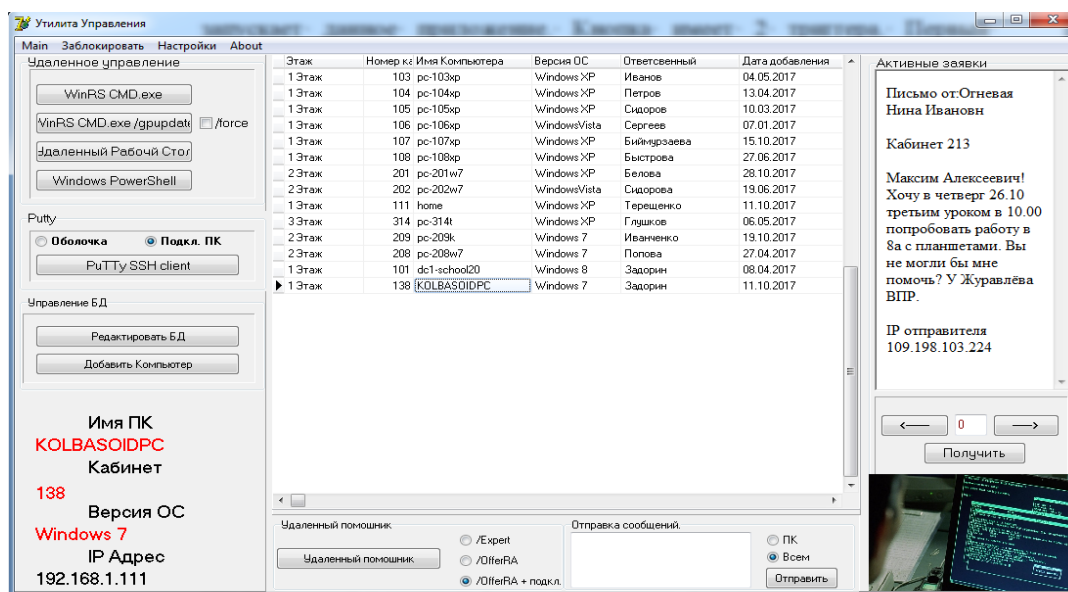


Рис 10. Основная форма программы

## 2.6.1 Функциональные кнопки блока «удаленное управление» и блока Putty

Кнопки блока «удаленное управление» выполняют аналогичные имени задачи на выбранной ИС. WinRS и PowerShell клиент службы удаленного управления службы «MS management» операционных систем семейства

Windows(xp и новее). Позволяет управлять (изменять настройки) удаленной ОС посредством командной строки. «WinRS cmd.exe», «PowerShell» запускает Командную строку Windows в разных функциональных оболочках. «WinRS gpupdate» запускает обновление групповых политик. Отмеченный триггер «/force» запускает gpupdate с ключом /force, выполняя принудительное обновление всех групповых политик. Кнопка «удаленный рабочий стол» выполняет подключение к выбранной ИС используя, клиент служб удаленных рабочих столов, для фактического управления рабочим столом удаленной ИС. Все перечисленные функциональные кнопки, при подключении используют реквизиты пары логин/пароль, указанных для ОС Windows, в одноименной функциональной кнопке «Логин/Пароль» для авторизации [Приложение 1].

Функциональные кнопка блока «Putty».

«Putty» универсальный клиент удаленного администрирования, поддерживающий telnet, rlogin, ssh протоколы. Предназначен для управления, в первую очередь для систем на базе ОС Linux и других ОС с поддержкой перечисленных протоколов. Функциональная кнопка запускает данное приложение. Кнопка имеет 2 триггера. Первый «оболочка» запускает клиент «Putty» в стандартной оболочке приложения. С возможностью подключаться к устройствам не из списка ИС БД. Второй триггер «Подкл. ПК» выполняет автоматическое подключение к ИС по протоколу SSH на стандартном порту, используя для авторизации пары логин/пароль Linux указанных в одноименной функциональной кнопке «Логин/Пароль». По умолчанию триггер установлен на «Подкл. ПК».

2.6.2 Блок управления БД, реквизитами авторизации, настройки электронной почты

Функциональная кнопка «Добавить компьютер» открывает форму добавления новой ИС в БД. Переходя на новую (пустую) строку для внесения информации о новой ИС.

Функциональная кнопка «Редактировать БД» вызывают ту же форму, но в форме «Редактирование БД» строки заполняются данными ИС выбранной в главной форме программы. Тем самым позволяя внести изменения в нужную ИС без поиска строки в базе данных.

Форма настройки электронной почты. На данной форме необходимо ввести сетевые настройки для получения сообщений электронной почты в приложении. (Рис 11). Для выполнения процедуры подключения к почтовому серверу [Приложение 2]

POP3Адрес – доменное имя для доступа к pop3 Серверу.

POP3Порт – порт, на котором работает POP3Сервер.

Версия шифрования подключения к POP3 – Выбор поддерживаемого типа шифрования для соединения с POP3Сервером.

Адрес эл. почты/пароль – реквизиты для подключения к POP3Серверу.

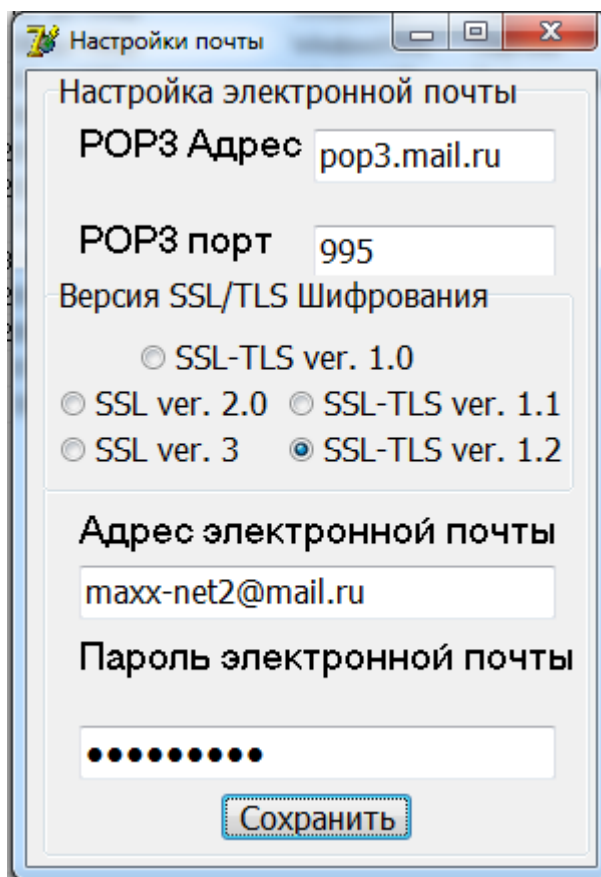
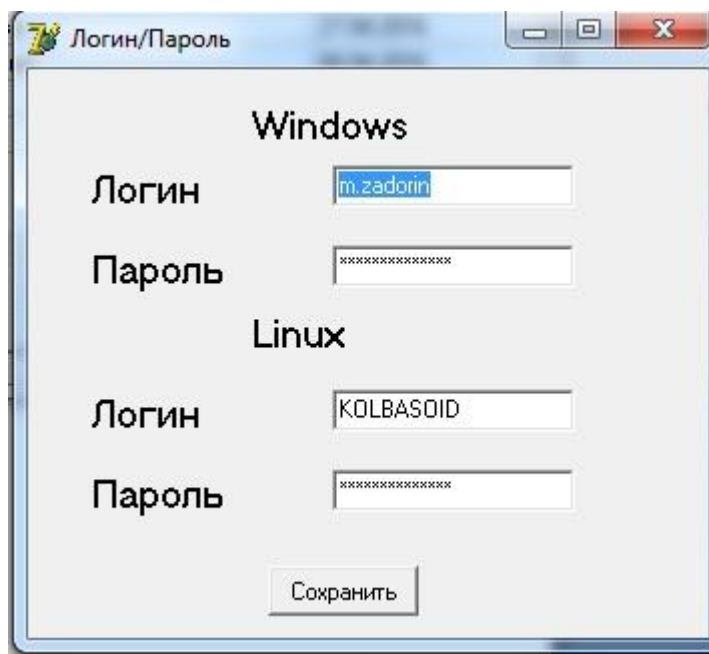


Рис 11. Форма настройка электронной почты

Функциональная кнопка «Логин/Пароль» позволяет внести регистрационные данные пары «Логин/Пароль» для авторизации при подключении к ИС. (Рис 12.)

The image shows a Windows-style dialog box titled "Логин/Пароль" (Login/Password). Inside the dialog, there are two sections: "Windows" and "Linux". Each section has two input fields: "Логин" (Login) and "Пароль" (Password). In the "Windows" section, the login field contains "m.zadorin" and the password field contains "xxxxxxxxxx". In the "Linux" section, the login field contains "KOLBASOID" and the password field contains "xxxxxxxxxx". At the bottom of the dialog, there is a button labeled "Сохранить" (Save).

*Рис 12. Форма ввода регистрационных данных логин/пароль*

Форма «Редактирование БД». Форма предназначена для добавления и редактирования базы данных ИС непосредственно из программы.

Поле формы «Информация о ПК» позволяет добавить новую ИС или отредактировать существующую ИС (Рис 13).

Редактирование БД

Информация о ПК

Этаж: 1 Этаж

Номер кабинета: 111

Имя Компьютера: home

Ответственный: Терещенко

Версия ОС: Windows XP

Дата Добавления: 11.12.2015

База данных

Этаж	Номер кабинета	Имя Компьютера	Ответственный	Версия ОС
2 Этаж	201	pc-201w7	Белова	Windows XP
2 Этаж	202	pc-202w7	Сидорова	Windows Vista
1 Этаж	203	KOLBASOIDh	Захаров	Windows 7
1 Этаж	111	home	Терещенко	Windows XP
3 Этаж	314	pc-314t	Глушков	Windows XP
2 Этаж	209	pc-209k	Иванченко	Windows 7
2 Этаж	208	pc-208w7	Попова	Windows 7
1 Этаж	101	dc1-school20	Задорин	Windows 8
1 Этаж	1	KOLBASOIDhomePC		Windows 7

Сохранить и выйти    Добавить ПК    < << >> >

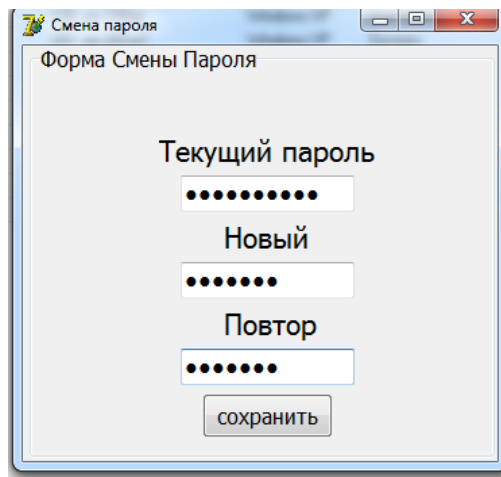
Рис 13. Форма редактирования базы данных программы

В зависимости от типа вводимых в БД данных строки «Этаж» и «Версия ОС» заполняются только выбором одного из нескольких вариантов. «Номер Кабинета» вводится типом данных целое число. «Имя Компьютера» и «Ответственный» заполняется типом данных текст. «Дата Добавления» тип дата в формате «дд.мм. гггг».

Поле формы «База Данных» отображает текущие ИС внесенные в БД позволяя выбрать нужную для редактирования ИС. Под формой «База Данных» располагаются функциональные кнопки. «Сохранить и выйти» сохраняет внесенные в БД изменения и закрывает форму «Редактирование БД». Функциональная кнопка «Добавить ПК» создает новую строку в конце БД для добавления новой ИС. Поле с кнопками навигации помогает перемещаться по БД (переход в конец/начало, следующая/предыдущая записи).

Форма смены пароля на доступ к программе. Форма позволяет сменить пароль на доступ к программе. Посредством ввода текущего, нового и повтор нового пароля. После нажатия кнопки сохранить, проверяется правильность ввода текущего пароля посредством чтения файла конфигурации и

дешифровки сохраненного там пароля, проверяется, верно, ли указан новый пароль, сравнивая эту строку со строкой «повтор». Если условия соблюдены новый пароль шифруется и записывается в файл конфигурации. (Рис 14.)



*Рис 14. Форма смены пароля пользователя*

Выход из программы и о программе. Выход из программы осуществляется либо нажатием на «крестик» в правом углу программы либо на нажатием, на меню «Main» - «Exit». В меню «About» информация о программе дате создания и авторе.

## ЗАКЛЮЧЕНИЕ

При тестировании и применении на практике программы Remote Network Tracer в средней образовательной школе №20 было выявлено, что возможности программы могут помочь в решении большинства повседневных задач, информационной поддержки пользователей учреждения, позволяя администратору ИТ службы удобно использовать возможности администрирования персональными компьютерами. Функционал программы предоставляет возможность службе поддержки независимо от присутствия пользователя на рабочем месте, подключаться к рабочим станциям, вносить изменения в настройки операционной системы, решать часть поставленных задач. Наличие системы отправки и получения информационных сообщений посредством Web-формы и установленного программного обеспечения на рабочих местах сотрудников, позволяют пользователю оперативно уведомить администратора о неисправности, а администратору максимально быстро отреагировать на заявку и при необходимости отправить сообщение пользователю о результатах проделанной работы. Благодаря наличию в программе Remote Network Tracer функции получения активных заявок непосредственно в основное окно приложения, специалист поддержки может удобно просматривать текущие заявки. Наличие базы данных позволяет хранить информацию об информационных системах, оперировать ей, получая актуальные данные, такие как: сетевое имя, ip адрес, ответственного сотрудника, и месторасположение рабочей станции. При необходимости проследовать в кабинет, где возникли неисправности. Наличие шифрования паролей пользователей помогает защитить конфиденциальные данные, и блокировать доступ в программу посторонним лицам.

Слабой стороной системы является невозможность запустить данное приложение «из коробки» т.е. сразу, без предварительной настройки программного обеспечения на рабочих станциях. Необходимо выполнить ряд действий от установки программного обеспечения до внесения изменений в



системные параметры операционной системы; иметь общую систему аутентификации пользователей и разрешения на доступ к компьютерам через локальную вычислительную сеть.

Совокупность всех перечисленных факторов объединенных в одном приложении позволяют повысить эффективность работы системного администратора, экономя его время при выполнении повседневных задач, добавляя удобство при работе с системами удаленного администрирования.

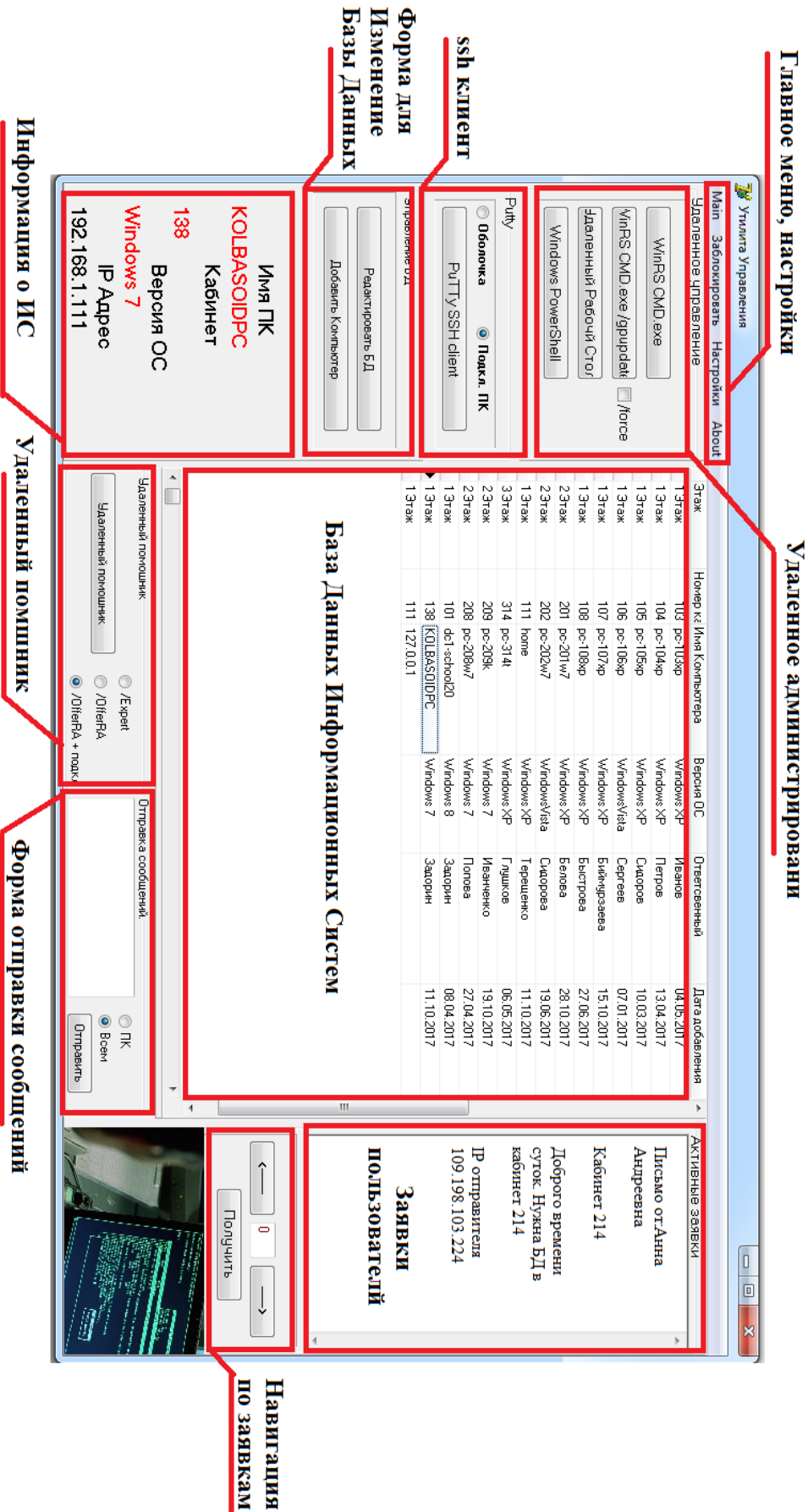
Таким образом, цель работы достигнута, задачи решены.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Архангельский А. Delphi 2006. Справочное пособие: Язык Delphi, классы, функции Win32 и .NET. М.:Бином, 2015 г. 1152 с.
2. Бунаков П., Лопатин А. Практикум по решению задач на ЭВМ в среде Delphi. Учебное пособие. М.:Инфра-м, 2017 г. 304 с.
3. Вернон В. Реализация методов предметно-ориентированного проектирования. М.:Диалектика, 2015 г. 688 с.
4. Дайн Д., Ландау Д., Хонг Д. Энциклопедия шаблонов веб-дизайна: шаблоны для создания непревзойденных веб-сайтов. М.: Вильямс, 2015 г. 1232 с.
5. Джордж К. Введение в системы баз данных. М.:Вильямс , 2017 г. 1328 с.
6. Дронов В., Прохоренок Н. HTML, JavaScript, PHP и MySQL. Джентльменский набор Web-мастера: М.:BHV, 2015 г. 768 с.
7. Емельянов С., Ватутин Э., Панищев В., Титов В. Процедурно-модульное программирование на Delphi. М.:Аргатакс-медиа, 2014 г. 352 с.
8. Зэндстра М. PHP: объекты, шаблоны и методики программирования. 4е издание. М.:Вильямс, 2015 г. 576 с.
9. Кнут Д. Искусство программирования, том 1. Основные алгоритмы. М.:Диалектика, 2015г. 720 с.
10. Кнут Д. Искусство программирования, том 4А. Комбинаторные алгоритмы, часть 1. М.:Вильямс, 2015 г. 960 с.
11. Майерс Г., Баджет Т., Сандлер К. Искусство тестирования программ, 3-е издание. М.:Вильямс, 2016 г. 272 с.

12. Неммет Э., Снайдер Г., Хейн Т., Уэйли Б. Unix и Linux: руководство системного администратора. Как установить и настроить Unix и Linux. М.:Диалектика, 2015 г. 1312 с.
13. Никсон Р. Создаем динамические веб-сайты с помощью PHP, MySQL, JavaScript, CSS и HTML5. СПб.:Питер, 2016 г. 768 с.
14. Осипов Д. Delphi. Программирование для Windows, OS X, iOS. СПб.:BHV, 2014 г. 464 с.
15. Осипов Д. InterBase и Delphi. Клиент-серверные базы данных: М.:ДМК-Пресс, 2015 г. 536 с.
16. Станек У. Microsoft Windows Server 2012 R2: хранение, безопасность, сетевые компоненты. Справочник. М.:BHV, 2015 г. 416 с.
17. Станек У. Справочник администратора. Internet Information Services (IIS) 7.0. М.:BHV, 2014 г. 528 с.
18. Сырых Ю. Современный веб-дизайн. Настольный и мобильный. М.:Диалектика, 2014 г. 384 с.
19. Фленов М. Библия Delphi. СПб.:BHV, 2014 г. 688 с.
20. Фуфаев Э., Фуфаев Д. Разработка и эксплуатация удаленных баз данных. Учебник. М.:Academia. 2014 г. 256 с.
21. Эванс Э. Предметно-ориентированное проектирование (DDD): структуризация сложных программных систем. М.:Вильямс, 2015 г. 448 с.
22. Эванс Э. Предметно-ориентированное проектирование (DDD): структуризация сложных программных систем. М.:Вильямс, 2015 г. 448 с.
23. «SENDMAIL INSTALLATION AND OPERATION GUIDE»  
[электронный ресурс]  
[http://www.opennet.ru/docs/RUS/sendmail\\_doc/sendmail-toc.html](http://www.opennet.ru/docs/RUS/sendmail_doc/sendmail-toc.html)

24. « Delphi компонент Indy: Indy Docs » [электронный ресурс]  
<http://www.indyproject.org/Socket/Docs/index.EN.aspx>
25. «Windows PowerShell Scripting» [электронный ресурс]  
<https://technet.microsoft.com/en-us/scriptcenter/dd742419.aspx>
26. «Русский блог команды разработчиков PowerShell»  
[электронный ресурс]  
[https://blogs.technet.microsoft.com/powershell\\_ru/](https://blogs.technet.microsoft.com/powershell_ru/)
27. «Основные команды в SSH используем PuTTY»  
[электронный ресурс] <http://eremeew.ru/page/osnovnye-komandy-v-ssh-ispolzuem-putty>
28. «Advanced Bash-Scripting Guide»[электронный ресурс]  
<http://rus-linux.net/MyLDP/BOOKS/abs-guide/flat/abs-book.html>
29. «Основы BASH» [электронный ресурс]  
<https://studylinux.ru/osnovy-bash.html>
30. «Материал из Википедии — свободной энциклопедии»,  
«База данных» [электронный ресурс]  
[https://ru.wikipedia.org/wiki/База\\_данных](https://ru.wikipedia.org/wiki/База_данных)



```

new 1 x
1 procedure TForm1.Button4Click(Sender: TObject);
2 var
3   p,i:Integer;
4   messtext1:string;
5 begin
6   WebBrowser1.Navigate2('about:blank');
7   p:=GetTickCount()+100;
8   repeat
9     Application.ProcessMessages;
10  until
11    GetTickCount() >= p;
12    IdPOP31.Host:=POP3Server;
13    IdPOP31.Port:=strtoint(POP3Port);
14    IdPOP31.Username:=Login;
15    IdPOP31.Password:=Password;
16    IdPOP31.IOHandler:= TIdSSLIOHandlerSocketOpenSSL.Create(IdPOP31);
17    IdPOP31.UseTLS:=utUseImplicitTLS;
18    fini := TIniFile.Create(ExtractFilePath(Application.ExeName)+'cfg.ini');
19    if fini.ReadString('Misc','SSLTLS','') = 'sslvTLsv1' then
20    begin
21      IdSSLIOHandlerSocketOpenSSL1.SSLOptions.Method:=sslvTLsv1;
22    end
23  else
24    if fini.ReadString('Misc','SSLTLS','') = 'sslvSSLv2' then
25    begin
26      IdSSLIOHandlerSocketOpenSSL1.SSLOptions.Method:=sslvSSLv2;
27    end
28  else
29    if fini.ReadString('Misc','SSLTLS','') = 'sslvSSLv3' then
30    begin
31      IdSSLIOHandlerSocketOpenSSL1.SSLOptions.Method:=sslvSSLv3;
32    end
33  else
34    if fini.ReadString('Misc','SSLTLS','') = 'sslvTLsv1_2' then
35    begin
36      IdSSLIOHandlerSocketOpenSSL1.SSLOptions.Method:=sslvTLsv1_2;
37    end;
38    IdPOP31.ConnectTimeout:=2000;
39    IdPOP31.Connect;
40    i:=IdPOP31.CheckMessages();
41    IdPOP31.Retrieve(i-strtoint(edit1.text),IdMessage1);
42    IdMessage1.SaveToFile('message.eml');
43    messtext1:=IdMessage1.Body.Text;
44    if (messtext1='') or (AnsiPos('multi-part message in MIME format',messtext1)>0) then
45      messtext1:='';
46    if (IdMessage1.ContentType='text/html') or (IdMessage1.ContentType='text/plain') then
47    begin
48      AppendToWB(WebBrowser1,messtext1);
49    end;
50  end;
51 end;

```

```

new 1 x
1 function Crypt(s:string;code:boolean):string;
2 var
3   i,Delta,Res:integer;
4 begin
5   Result:='';
6   for i:=1 to Length(s) do
7   begin
8     Delta:=((i xor Pas) mod (256-32));
9     if code then
10      Res:=((ord(s[i])+Delta) mod (256-32))+32
11    else
12    begin
13      Res:=ord(s[i])-Delta-32;
14      if Res<32 then
15        Res:=Res+256-32;
16      end;
17      Result:=Result+chr(Res);
18    end;
19  end;
20
21
22 procedure TForm1.FormCreate(Sender: TObject);
23 var
24   fini: TIniFile;
25 begin
26   pas:=*****;
27   fini := TIniFile.Create(ExtractFilePath(Application.ExeName)+'cfg.ini');
28   WindowsName := fini.ReadString('Main','WindowsName','');
29   WindowsPass := Crypt(fini.ReadString('Main','WindowsPass',''),false);
30   LinuxName := fini.ReadString('Main','LinuxName','');
31   Linuxpass := Crypt(fini.ReadString('Main','LinuxPass',''),false);
32   POP3server := fini.ReadString('POP3','POP3Server','');
33   POP3Port := fini.ReadString('POP3','POP3Port','');
34   Login := fini.ReadString('POP3','Login','');
35   Password := crypt(fini.ReadString('POP3','Password',''),false);
36   fini.Free;
37 end;
38
39
40 procedure TfREG.BitBtn1Click(Sender: TObject);
41 begin
42   fini:=TIniFile.Create(ExtractFilePath(Application.ExeName)+'cfg.ini');
43   fini.WriteString('Main','WindowsName',Edit1.Text);
44   fini.WriteString('Main','WindowsPass',Crypt(edit2.text,true));
45   fini.WriteString('Main','LinuxName',Edit3.Text);
46   fini.WriteString('Main','LinuxPass',Crypt(Edit4.Text,true));
47   fini.Free;
48   close;
49 end;
50

```

**Алгоритм Шифрования**

**Дешифровка Текста**

**Шифрование Текста**